UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/643,758	08/18/2003	Kitrick Sheets	1376.729US1	3924

21186 7590 02/06/2007

SCHWEGMAN, LUNDBERG, WOESSNER & KLUTH, P.A.
P.O. BOX 2938
MINNEAPOLIS, MN 55402

EXAMINER	
TSAI, SHENG JEN	
ART UNIT	**PAPER NUMBER**
2186	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	02/06/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	10/643,758	SHEETS ET AL.	
	Examiner	Art Unit	
	Sheng-Jen Tsai	2186	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 December 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-22* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-22* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *18 August 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

1. This Office Action is taken in response to Applicants' Amendments and Remarks filed on December 14, 2006 regarding application 10,643,758 filed on August 18, 2003.

2. Claims 1-2, 4, 6, 8-9 and 11 have been amended.

Claims 18-22 have been added.

Claims 1-22 are pending for consideration.

3. ***Response to Remarks and Amendments***

Applicants' amendments and remarks have been fully and carefully considered, with the Examiner's responses set forth below.

Independent claims 1, 4, 6, 9 and 11 have been amended to include the new limitation of "**translating the virtual memory reference into a physical address in the application virtual address space using the RTT on the local node, if the node number is not the local number and the remote translation is enabled**; ..."

This newly added limitation raises question under 35 U.S.C. 112, first paragraph for not being support by the written description of Applicants' disclosure (Sheets et al., US Patent Application Publication 2005/0044340, hereafter referred to as the Application), as explained below:

First, paragraph [0006] of the Application recites "The method includes providing the virtual memory address at a source node, determining that the virtual memory address is to be sent to a remote node, sending the virtual memory address to the remote node, and translating the virtual memory address on the remote node into a physical memory address using a remote-translation table (RTT). The RTT contains

translation information for an entire virtual memory address space associated with the remote node."

Here, it recites sending the virtual memory address to the remote node, and then translating the virtual memory address on the remote node into a physical memory address using a remote-translation table (RTT). Thus it is clear from the description that the translation is performed using the RTT at the remote node, not using the RTT at the local node.

Second, paragraph [0007] of the Application recites "The method includes providing a virtual memory address on a local node by using a virtual address of a load or a store instruction, identifying a virtual node associated with the virtual memory address, and determining if the virtual node corresponds to the local node. If the virtual node corresponds to the local node, then the method includes translating the virtual memory address into a local physical memory address on the local node. If, instead, the virtual node corresponds to a remote node, then the method includes sending the virtual memory address to the remote node, and translating the virtual memory address into a physical memory address on the remote node."

Again, it recites sending the virtual memory address to the remote node, and then translating the virtual memory address on the remote node into a physical memory address using a remote-translation table (RTT). Thus it is clear from the description that the translation is performed using the RTT at the remote node, not using the RTT at the local node.

Third, paragraph [0025] of the Application recites "If remote translation is enabled, however, and the virtual address is for a remote node (as determined by comparing the virtual node field of the virtual address with the value representing the local virtual node number), then a remote address translation mechanism is used, as follows. A physical node number is determined by adding the virtual node field of the virtual address to a physical base node. The virtual node number is also checked against a limit value, to ensure that the request is within allowable bounds. The remainder of the virtual address forms a virtual offset, which is sent with the memory request to the destination physical node. A "global address space identifier" (GASID) is also looked up for the local processor and sent with the request. The GASID and the upper portion of the virtual address are used to index into a remote translation table (RTT) at the destination node, to produce a physical page number at the remote node."

Once more, it describes that a RTT at the remote/destination node is used to generate a physical address at the remote node if remote translation is enabled and the virtual address is for a remote node.

Therefore, the newly added limitation of "**translating the virtual memory reference into a physical address in the application virtual address space using the RTT on the local node, if the node number is not the local number and the remote translation is enabled**" not only is not supported by, but also contradictory to, the written description of the Application.

Further, the references relied on in the previous Office Action (Schimmel, US 6,105,113 and Scott et al., US 6,925,547) teach the particular method of translating

virtual address destined for a remote node using the RTT at the remote node that is consistent with the written description of Applicants' disclosure.

For example, Scott et al. teach that [According to one aspect of the invention, a method of performing remote address translation in a multiprocessor system includes determining a virtual address at a local node, accessing a local connection table at the local node to produce a system node identifier for a remote node, communicating the virtual address to the remote node, and translating the virtual address to a physical address at the remote node. The translation may include matching the virtual address with an entry of a translation-lookaside buffer at the remote node, and may also use a remote address space number as a qualification or validation for the match. According to another aspect of the invention, a mechanism for performing a memory access operation in a multiprocessor system uses a communication engine located in a local processing element node and a translation-lookaside buffer located in a remote processing element node. The communication engine is programmable by a user process to perform a user-level memory access operation using a user-specified virtual address. The translation-lookaside buffer receives the virtual address from the communication engine and translates the virtual address to a physical address, wherein the physical address is used to perform the memory access operation (column2, 65-67 and column 3, lines 1-21)].

Therefore, the Examiner's position regarding the merits of patentability of all claims, including both the double patenting rejections as well as rejections under 35 U.S.C. 103(a), remains the same as stated in the previous Office Action.

Double Patenting

4. The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970); and, *In re Thorington,* 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

5. Claims 1-2, 4, 6, 8-9 and 11 are rejected under the judicially created doctrine of anticipation-type double patenting as being anticipated by claims 1-35 of US patent **6,922,766** (Scott, "Remote Translation Mechanism for a Multi-Node System"), as shown in the following table. Although not all of the conflicting claims are exactly identical, they are extremely similar and are not patentably distinct from each other as explained in the "explanation" column of the table below:

6,922,766	10/643,758	EXPLANATION
1, 2, 7 and 12	1	Both describe similar components with similar features/functions: virtual address space translation, Remote Translation Table (RTT), the virtual address generated includes a node number field
15	2	Both recite that the local address space is provided by a Translation Look-aside Buffer (TLB)
2 and 25-29	4	Both describe similar components with similar features/functions: processor, memory, RTT, address translation, the virtual address generated includes a node number field
1, 2, 7 and 12	6	Both describe similar components with similar features/functions: virtual address space translation, Remote Translation Table (RTT), the virtual address generated includes a node number field
15	8	Both recite that the local address space is provided by a Translation Look-aside Buffer (TLB)
2 and 25-29	9	Both describe similar components with similar features/functions: processor, memory, RTT, address translation, the virtual address generated includes a node number field
1-3	11	Both describe similar components with similar features/functions: network, virtual address space translation, Remote Translation Table (RTT), the virtual address generated includes a node number field

6. Claims 1-2, 4, 6, 8-9 and 11 are provisionally rejected under the judicially created doctrine of anticipation-type double patenting as being anticipated by claims 1-15 of co-pending US patent application No. **10/643,588** (Sheets, "Sharing Memory within an Application Using Scalable Hardware Resources"), as shown in the following table. Although not all of the conflicting claims are exactly identical, they are extremely similar and are not patentably distinct from each other as explained in the "explanation" column of the table below:

6,922,766	10/643,758	EXPLANATION
1, 3	1	Both describe similar components with similar features/functions: virtual address space translation, Remote Translation Table (RTT), the virtual address generated includes a node number field
1	2	Both recite that the local address space is provided by a Translation Look-aside Buffer (TLB)
6, 8	4	Both describe similar components with similar features/functions: processor, memory, RTT, address translation, the virtual address generated includes a node number field
1, 3	6	Both describe similar components with similar features/functions: virtual address space translation, Remote Translation Table (RTT), the virtual address generated includes a node number field
1, 6	8	Both recite that the local address space is provided by a Translation Look-aside Buffer (TLB)
6, 8	9	Both describe similar components with similar features/functions: processor, memory, RTT, address translation, the virtual address generated includes a node number field
1, 6, 11	11	Both describe similar components with similar features/functions: network, virtual address space translation, Remote Translation Table (RTT), the virtual address generated includes a node number field

Claim Rejections - 35 USC § 112

7. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the

art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

8. Claims 1-22 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Independent claims 1, 4, 6, 9 and 11 have been amended to include the new limitation of "**translating the virtual memory reference into a physical address in the application virtual address space using the RTT on the local node, if the node number is not the local number and the remote translation is enabled**; ..."

This newly added limitation raises question under 35 U.S.C. 112, first paragraph for not being support by the written description of Applicants' disclosure (Sheets et al., US Patent Application Publication 2005/0044340, hereafter referred to as the Application), as explained below:

First, paragraph [0006] of the Application recites "The method includes providing the virtual memory address at a source node, determining that the virtual memory address is to be sent to a remote node, sending the virtual memory address to the remote node, and translating the virtual memory address on the remote node into a physical memory address using a remote-translation table (RTT). The RTT contains translation information for an entire virtual memory address space associated with the remote node."

Here, it recites sending the virtual memory address to the remote node, and then translating the virtual memory address on the remote node into a physical memory address using a remote-translation table (RTT). Thus it is clear from the description that the translation is performed using the RTT at the remote node, not using the RTT at the local node.

Second, paragraph [0007] of the Application recites "The method includes providing a virtual memory address on a local node by using a virtual address of a load or a store instruction, identifying a virtual node associated with the virtual memory address, and determining if the virtual node corresponds to the local node. If the virtual node corresponds to the local node, then the method includes translating the virtual memory address into a local physical memory address on the local node. If, instead, the virtual node corresponds to a remote node, then the method includes sending the virtual memory address to the remote node, and translating the virtual memory address into a physical memory address on the remote node."

Again, it recites sending the virtual memory address to the remote node, and then translating the virtual memory address on the remote node into a physical memory address using a remote-translation table (RTT). Thus it is clear from the description that the translation is performed using the RTT at the remote node, not using the RTT at the local node.

Third, paragraph [0025] of the Application recites "If remote translation is enabled, however, and the virtual address is for a remote node (as determined by comparing the virtual node field of the virtual address with the value representing the

local virtual node number), then a remote address translation mechanism is used, as follows. A physical node number is determined by adding the virtual node field of the virtual address to a physical base node. The virtual node number is also checked against a limit value, to ensure that the request is within allowable bounds. The remainder of the virtual address forms a virtual offset, which is sent with the memory request to the destination physical node. A "global address space identifier" (GASID) is also looked up for the local processor and sent with the request. The GASID and the upper portion of the virtual address are used to index into a remote translation table (RTT) at the destination node, to produce a physical page number at the remote node."

Once more, it describes that a RTT at the remote/destination node is used to generate a physical address at the remote node if remote translation is enabled and the virtual address is for a remote node.

Therefore, the newly added limitation of "**translating the virtual memory reference into a physical address in the application virtual address space using the RTT on the local node, if the node number is not the local number and the remote translation is enabled**" not only is not supported by, but also contradictory to, the written description of the Application.

All dependent claims are rejected by virtue of their dependency from the respective independent claim.

Claim Rejections - 35 USC § 103

9. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 103 that form the basis for the rejections under this section made in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

10. Claims 1-22 are rejected under 35 U.S.C. 103(a) as being unpatentable over Schimmel (US 6,105,113), and in view of Scott et al. (US 6,925,547).

Please refer to "***Response to Remarks and Amendments***" presented earlier in this Office Action regarding teachings of the newly added limitations by the references.

As to claim 1, Schimmel discloses **a method of accessing shared memory in a computer system having a plurality of nodes** [System and Method for Maintaining Translation Look-aside Buffer (TLB) Consistency (title); figure 3 shows a Distributed Shared Memory (DSM) system], **including a first node** [any node in figure 3 may be the first node], **wherein each node includes a processor and local memory** [figure 3], **the method comprising**:

distributing an application across the plurality of nodes [the present invention can be implemented on any computer system that employs virtual memory. Thus, the present invention can be implemented in both uni-processor environments and multiple processor environments. The present invention is especially useful in shared memory, multi-processor systems where page migration occurs. Shared memory systems that benefit from the present invention include centralized shared memory systems, such as, symmetric multiple processor (SMP) systems and distributed shared memory (DSM) systems. The present invention can be employed to maintain consistency for any number of TLBs in a system (column 4, lines 32-43)];

building an application virtual address space [in a virtual memory scheme, each process that is allocated a block of physical memory is also provided with a set of translations for translating virtual addresses to assigned physical addresses of the allocated block ... (column 1, lines 24-46)], **wherein building an application virtual address space includes**:

building a local virtual address space for the application in each of the plurality of nodes, wherein the local virtual address space translates a virtual address generated by the application executing on that node to a physical address in local memory for that node [by distributing physical or main memory 328-342 throughout DSM 310, each processing node 350-364 can include a portion of main memory. This physical proximity between processor and memory reduces memory latency with respect to the processor and memory within a processing node. DSM 310 is preferably configured so that data which is accessed most frequently by a particular processing node is placed in the portion of main memory within the processing node. If that data is subsequently accessed more frequently by a processor in another processing node, the data is migrated, or moved, to a portion of main memory within the other processing node (column 7, lines 30-42); in operation, when a CPU requires a physical memory address that is associated with a virtual memory address, the CPU first searches the virtual address tag of the TLB table. If a valid translation is not found in the TLB table, the translation is retrieved from a cache or from main memory and a copy of the translation is placed in the TLB table (column 4, lines 8-13)]; **and**

the virtual address generated by the application executing on the node includes a node number [this limitation is taught by Scott et al., see below]; **and exporting the local virtual address space for each node to a Remote Translation Table (RTT) associated with that node** [in a virtual memory scheme, each process that is allocated a block of physical memory is also provided with a set of translations for translating virtual addresses to assigned physical addresses of the allocated block. Each set of translations can be stored in, for example, a page table. A page table can be associated with a specific user or shared by multiple users (column 1, lines 24-31); figure 8 shows the RTT table; column 3, lines 42-62]; **wherein exporting includes requesting, at a processor within each node, that the operating system load the RTT from the local address space of its respective node and requesting that the operating system enable remote translation** [whenever a translation in TLB 718 is to be invalidated, CPU 714 or the operating system must be interrupted in order to execute the invalidation (column 9, lines 65-67); a copy of the translation is also placed in TLB 718. Later, when CPU 714 requires a translation, CPU 714 or an operating system searches TLB 718. If the translation is not found in TLB 718 (i.e., a TLB "miss"), the desired translation can be loaded from the page tables in memory by hardware, software, firmware, or any combination thereof (column 9, lines 22-28); an operating system that controls processor and cache node 410 can map virtual memory, as discussed above in FIGS. 5 and 6, where virtual memory addresses 512 are mapped to physical addresses 516 (column 11, lines 52-54); in step 912, the operating system generates virtual memory address-to-physical memory address translations for

the mapped data. For example, a page table, such as page table 610 can be generated for a process that is provided with mapped memory. The page table can be for the exclusive use of one user or can be shared by multiple users (column 11, lines 59-64); the operation can be performed solely by the CPU, solely by an operating system (not shown) that controls the CPU or by a combination of the CPU and the operating system (column 10, lines 39-42); the PTE for page 0x10 can be found by simply looking at index 0x10 in the page table array. The starting address of the array itself is maintained by the operating system in such a way that it is easy to find (column 8, lines 40-44)]; **and**

performing a memory reference to a memory location in the application virtual address space, if the node number is not local node number; wherein performing a memory reference to a memory location in the application virtual address space includes translating the node number of the application virtual address into a node address associated with the first node and translating bits of the application virtual address using the RTT associated with the first node [in operation, when a CPU requires a physical memory address that is associated with a virtual memory address, the CPU first searches the virtual address tag of the TLB table. If a valid translation is not found in the TLB table, the translation is retrieved from a cache or from main memory and a copy of the translation is placed in the TLB table (column 4, lines 8-13); column 7, lines 50-67; column 8, lines 1-67].

Regarding claim 1, Schimmel does not explicitly mention that **the virtual address generated by the application executing on the node includes a node number**.

However, it should be noted that Schimmel's invention is directed toward virtual memory address translation in a Distributed Shared Memory (DSM) system where each processing node has a portion of distributed shared memory [figure 3, column 6, lines 63-65]. In such a system, when a processing node tries to access a memory portion that is not located at its own node it must find out which processing node the target memory is located. Thus, the node number, or the identity of the node having the target memory is implied in Schimmel's invention, otherwise a processing node would not be able to access memory portions that are distributed to other processing nodes.

Further, Scott et al. disclose in their invention "Remote Address Translation in a Multiprocessor System" a method of performing remote, virtual address translation where **the virtual address generated by the application executing on the node includes a node number** [According to one aspect of the invention, a method of performing remote address translation in a multiprocessor system includes determining a virtual address at a local node, accessing a local connection table at the local node to produce a system node identifier for a remote node, communicating the virtual address to the remote node, and translating the virtual address to a physical address at the remote node. The translation may include matching the virtual address with an entry of a translation-lookaside buffer at the remote node, and may also use a remote

address space number as a qualification or validation for the match (column 2, lines 65-67; column 3, lines 1-9)].

Therefore, it would have been obvious for one of ordinary skills in the art at the time of Applicants' invention to recognize that the node identifier (i.e., node ID number) is inherently needed for virtual address translation in a multi-node, distributed shared memory environment, as implied by Schimmel and explicitly taught by Scott et al., thus lacking patentable significance.

As to claim 2, Schimmel teaches that **the local address space is read from a Translation Look-aside Buffer (TLB)** [figure 8; column 3, lines 42-62].

As to claim 3, In order for the remote translation mechanism disclosed by Schimmel to work and function properly, it is inherent that the RTT at all the nodes be initialized and synchronized first before any reference to a memory location resides at a remote node can be served. Without the initialization and synchronization, the RTT may not have the correct information to reach the correct memory location.

As to claim 4, Schimmel teaches **a system comprising**:

a plurality of nodes [figure 3], **each node including**:

one or more processors [figure 3];

a memory [figure 3]; **and**

a memory controller operatively coupled to the memory and the one or more processors [figure 3 shows the cache coherency directory], **wherein the memory controller includes a Remote Translation Table (RTT)** [figure 8 shows the RTT table; column 3, lines 42-62], **wherein the RTT translates a virtual address received**

as part of a memory request received from another node into a memory request with physical addresses into the memory on the node associated with the RTT [in operation, when a CPU requires a physical memory address that is associated with a virtual memory address, the CPU first searches the virtual address tag of the TLB table. If a valid translation is not found in the TLB table, the translation is retrieved from a cache or from main memory and a copy of the translation is placed in the TLB table (column 4, lines 8-13); column 7, lines 50-67; column 8, lines 1-67]; **further wherein the RTT is initialized upon the start of a process associated with an application by building virtual to physical address translations for local virtual address space in the node corresponding to the application** [In order for the remote translation mechanism disclosed by Schimmel to work and function properly, it is inherent that the RTT at all the nodes be initialized and synchronized first before any reference to a memory location resides at a remote node can be served. Without the initialization and synchronization, the RTT may not have the correct information to reach the correct memory location], **wherein a virtual address includes a node number of the node** [refer to the explanation provided in "As to claim 1"], **and exporting the virtual to physical address translations for the local virtual address space from the node to the Remote Translation Table (RTT) associated with that node** [in a virtual memory scheme, each process that is allocated a block of physical memory is also provided with a set of translations for translating virtual addresses to assigned physical addresses of the allocated block. Each set of translations can be stored in, for example, a page table. A page table can

be associated with a specific user or shared by multiple users (column 1, lines 24-31); figure 8 shows the RTT table; column 3, lines 42-62].

As to claim 5, In order for the remote translation mechanism disclosed by Schimmel to work and function properly, it is inherent that the RTT at all the nodes be initialized and synchronized first before any reference to a memory location resides at a remote node can be served. Without the initialization and synchronization, the RTT may not have the correct information to reach the correct memory location.

As to claim 6, refer to "As to claim 1."

As to claim 7, refer to "As to claim 3."

As to claim 8, refer to "As to claim 2."

As to claim 9, refer to "As to claim 1" and "As to claim 4."

As to claim 10, refer to "As to claim 3."

As to claim 11, refer to "As to claim 1" and "As to claim 4." Further, figure 3 shows a network as part of the system.

As to claim 12, Schimmel teaches that **requesting the operating system enable remote translation passes control of the RTT to the operating system** [whenever a translation in TLB 718 is to be invalidated, CPU 714 or the operating system must be interrupted in order to execute the invalidation (column 9, lines 65-67); a copy of the translation is also placed in TLB 718. Later, when CPU 714 requires a translation, CPU 714 or an operating system searches TLB 718. If the translation is not found in TLB 718 (i.e., a TLB "miss"), the desired translation can be loaded from the page tables in memory by hardware, software, firmware, or any combination thereof

(column 9, lines 22-28); an operating system that controls processor and cache node 410 can map virtual memory, as discussed above in FIGS. 5 and 6, where virtual memory addresses 512 are mapped to physical addresses 516 (column 11, lines 52-54); in step 912, the operating system generates virtual memory address-to-physical memory address translations for the mapped data. For example, a page table, such as page table 610 can be generated for a process that is provided with mapped memory. The page table can be for the exclusive use of one user or can be shared by multiple users (column 11, lines 59-64); the operation can be performed solely by the CPU, solely by an operating system (not shown) that controls the CPU or by a combination of the CPU and the operating system (column 10, lines 39-42); the PTE for page 0x10 can be found by simply looking at index 0x10 in the page table array. The starting address of the array itself is maintained by the operating system in such a way that it is easy to find (column 8, lines 40-44)].

As to claim 13, Schimmel teaches that **passing control of the RTT to the operating system causes the operating system to maintain coherency of the RTT** [a system and method for maintaining consistency between translational look-aside buffers (TLB) and page tables (abstract); one problem that confronts both TLBs and caches is maintaining consistency of data that is stored in more than one location ... Thus, the operating system updates the PTE to reflect the new physical location of the data (column 2, lines 12-25); cache consistency can be maintained between data cached in cache 416 -and data stored in main memory 810 by a variety of consistency techniques. For example, cache consistency can be maintained with an optional cache

consistency directory 814 in main memory 810. Alternatively, cache consistency can be maintained by a snooping protocol implemented within cache controller 812 which snoops bus 816 for broadcast messages (column 10, lines 15-22); the operation can be performed solely by the CPU, solely by an operating system (not shown) that controls the CPU or by a combination of the CPU and the operating system (column 10, lines 39-42)].

As to claim 14, refer to "As to claim 12."

As to claim 15, refer to "As to claim 13."

As to claim 16, refer to "As to claim 12."

As to claim 17, refer to "As to claim 13."

As to claim 18, Scott et al. teach that **the method of claim 1, wherein requesting the operating system enable remote translation handles requests to changes the application virtual address space configuration on a node-local basis** [column2, 65-67 and column 3, lines 1-21], **wherein handling requests includes disallowing an attempt to modify the application virtual address space outside scope of the local node** [8=TRANS_ERR_SRC (translation error for source connection, set by an error response due to a special remote TLB entry, or by local OS); 9=TRANS_ERR_DEST (translation error for destination connection); and 10=HW_ERROR (hardware error) (column 12, lines 32-37)].

Claims 19-22 are rejected by the same reason as applied to the rejection of claim 18.

Conclusion

11. Claims 1-22 are rejected as explained above.

12. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

13. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Sheng-Jen Tsai whose telephone number is 571-272-4244. The examiner can normally be reached on 8:30 - 5:00.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Matthew Kim can be reached on 571-272-4182. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Sheng-Jen Tsai
Examiner
Art Unit 2186

January 26, 2007

PIERRE BATAILLE
PRIMARY EXAMINER
2/1/07